

By: Rbx2007Popa, Linux Editor

Vim Text Editor

Programmer's best editor

When it comes to text editors, one cannot complain s/he hasn't enough options to choose from. A quick search on [Wikipedia](#) would lead you to an impressive number of currently 34 such editors. From the terminal-based Joe or Nano up to more sophisticated stuff such as Emacs or gPHPedit, you'll find a large suite of editors, each of them designed to better suite one's needs. A more special case among these editors would be **Vim**. Vim can be a very powerful tool in the hands of a skilled programmer. I said it right...it's 'programmer', since Vim addresses mostly this class of users also being called "programmers' editor". This is not a strict rule though. A common user could try dealing with Vim, but he would have to do a little study before and I'm not sure up to what extent someone would start learning how to use a complicated tool for mastering ordinary things when simpler solutions are available. However, for those interested in Vim, its creators also released a manual to help them a bit. I could also count here the Vim tutorial for beginners, which can be opened by typing *vimtutor* on the Unix command line. If in the beginning the learning process is slow, once the user gets the basics, progress should come very quickly and one's editing becomes significantly more efficient. With a special and individualized GUI, Vim does not match the patterns of other common editors. It can be compared with its ancestor, **vi**, up to one point; after that, the features offered by Vim are harder to find in others. **vi** editor was originally written by Bill Joy in 1976 for an early BSD release. **vi** benefits of more editing modes, therefore it's a modal editor and it would then assign different meanings to buttons or keystrokes, depending on the active editing mode. The fact that it is able to start up much faster than versions of Emacs and uses less memory made **vi** popular among many Unix users. According to a poll from 1991, half the people in question answered they use **vi**, which is a really high percentage. Now Vim, as derived from **vi**, has inherited all of these **vi** functionalities but also came with new features. Standing for **Vi Improved**, Vim includes, among many others, features such as command history and editing, filename completions, multiple edit buffers, syntax highlighting, mouse support, graphical versions, visual mode, split window (both horizontal and vertical) and many new editing commands and it has become the standard version of **vi** on most Linux systems. Many of Vim's features are very helpful in editing program source code, fact that makes it highly beloved among programmers and developers. Here I could take as a small example the syntax highlighting, which is available for a very wide range of file types. An aspect that is worth mentioning would be that, even though Vim is a powerful text editor, it is not a word processor too. As I've said before, Vim is a modal editor. Therefore, it has several modes which differ in their functionality. The default mode is the *Normal* one, where the user can type in editor commands such as cursor movement, text deletion a.s.o. Another aspect I've found interesting about Vim would be the fact that all things can be easily done by making use of the keyboard only, saving a lot of time and providing a faster way for accomplishing different tasks. The *Visual* mode looks like the *Normal* one, but the movement commands extend highlighted text. The highlighting can be set by characters, lines, or a block of text. The *Insert* mode is used for typing text in. A mode similar to this one is also the *Replace* mode. The difference between the two modes is that in *Replace* each character typed overwrites an existing character in the buffer. In the *Select* mode you can highlight text. The highlighted area will be automatically replaced with a character if it's typed, which will cause Vim to enter insert mode. Besides these "standard" modes there are also some hybrid ones such as the *Insert Visual*, *Insert Selector* *Insert Normal*. And if you think these should be enough, well, they aren't. To complete the set, Vim's creators added the *Command-line*, the *Ex* and the *Evim* modes. Another feature Vim is famous for would be its highly customization power. The user has the possibility of customizing the basic interface or define their own key mappings or even call internal or user defined functions. As it benefits of its own scripting language, there are also many plugins written in this language available for extending/personalizing vim. **The Good** Vim is a professional text editor targeting mainly programmers, but it can be also used by any other type of user. It received the *best text editor* award from readers of the Linux Journal magazine in 2001, 2002, 2003, 2004 and 2005, which I think tells a lot about its efficiency. **The Bad** The bad and ugly part about Vim would be for most users the fact that they must do the study first in order to

be able to use it. However, once you've managed that, the benefits are enormous and I think it's definitely worth all the effort. ***The Truth*** Vim is a multi-platform text editor, which means you can use it regardless of the operating system that powers your machine. Fully compatible too **vi**, Vim provides all the functionality of a regular text editor, such as spell checking, Unicode support, possibility of undo/redo the most recent actions, integrated help but it has also some features of its own which make it a powerful and reliable tool. *To get a better idea on what Vim is, you can also check out the pictures below.*